IN THE CLAIMS:

1. (currently amended) A method of decoding errors occurring in data stored in memory, comprising:

applying data to be stored in a buffer memory as a plurality of data words to a generator matrix to generate parity check bits;

storing the data words in a plurality of data buffer locations in the buffer memory and the parity check bits in one or more parity check buffer locations in the buffer memory, the parity check buffer locations being different locations than the data buffer locations that contain the data;

reading the stored data words and the parity check bits from the respective data buffer and parity check buffer locations;

regenerating the parity check bits; and

producing from the stored and regenerated parity check bits a result that is usable to directly identify

i. the data buffer location that contains a data word ~~that contains~~ with an erroneous bit, and

ii. the position of the erroneous bit in the data word contained in the identified data buffer location.

2. (original) The method of claim 1 wherein the result is in the form of a syndrome.

3. (previously amended) The method of claim 1 wherein the result identifies an address of the data buffer location.

4. (cancel)

5. (currently amended) The method of claim 1 wherein the generator matrix comprises a data portion and a parity check generation portion, and the parity check generation portion comprises rows of bits corresponding to binary representations of the data buffer locations used to store the data words.

6. (original) The method of claim 5 wherein the parity check generation portion of the generator matrix comprises columns of bit sequences usable to select combinations of data bits for parity check bit generation, each column corresponding to a different parity check bit.

7. (original) The method of claim 6 wherein the generator matrix has a minimum distance of three and describes an error correction code with single-bit error correction capability.

8. (previously amended) The method of claim 7 wherein one of the columns is usable to generate a parity check bit for the parity check bits that correspond to the data.

9. (original) The method of claim 7 wherein one of the columns is usable to select data parity in producing a parity check bit.

10. (original) The method of claim 6 wherein the generator matrix has a minimum distance of four and describes an error correction code with single-bit error correction capability.

11. (original) The method of claim 10 wherein the columns comprise a column to select data parity in producing a parity check bit and a column to generate a parity check bit for the parity check bits that correspond to the data.

12. (previously amended) The method of claim 6 wherein the generator matrix has a minimum distance of five and describes an error correction code with a double-bit error correction capability, and wherein producing comprises:

producing from the stored and regenerated parity check bits a result that is usable to directly identify the positions of two erroneous bits of the data word contained in the identified data buffer location.

13. (original) The method of claim 12 wherein the rows of the parity check generation portion comprise bits corresponding to a first field element and a second field element, and wherein the second field element has the same properties as the first field element.

14. (original) The method of claim 13 wherein the second field element is generated from the first field element.

15. (original) The method of claim 13 wherein the second field element is generated from the first field element by a cyclic rotation of bits in the first field element.

16. (original) The method of claim 13 wherein the first and second field elements are field elements of a Galois field of $GF(2^p)$, where p is an integer.

17. (original) The method of claim 16 wherein the first element comprises a binary representation β^k and the second element comprises a binary representation β^{sk} where k is an integer in a range of 1 to 2^p-1 and s does not divide 2^p-1.

18. (original) The method of claim 17 wherein the integer p is equal to 14.

19. (original) The method of claim 18 wherein s is equal to 5.

20. (previously amended) The method of claim 17 wherein the two erroneous bits are associated with ones of the first and second elements, and the first and second elements provide the positions of the two erroneous bits.

21. (previously amended) The method of claim 20 wherein the first element comprises a binary representation β^k and the second element comprises a binary representation β^{-k} where k is an integer in a range of 1 to 2^p-1.

22. (original) The method of claim 14 wherein the first element comprises a normal basis representation of a binary number.

23. (previously amended) An encoding method comprising:

applying data to be stored in a buffer memory to a generator matrix as a plurality of data words to generate a set of parity check bits, the generator matrix comprising a data portion and a parity check generation portion, and the parity check generation portion comprises rows of bits corresponding to binary representations of respective data buffer locations to be used to store the data words;

storing the data in the buffer memory at the data buffer locations; and

storing the set of parity check bits in one or more parity check locations of the buffer memory, the parity check locations being different locations than the data buffer locations that contain the data.

24. (currently amended) A data storage system comprising:

a storage medium;

a controller coupled to the storage medium; and

a buffer memory coupled to the storage medium and the controller for storing data to

be written to the storage medium and data read from the storage medium;

wherein the controller is operable to perform the following steps:

applying data to be stored in a buffer memory as a plurality of data words to a generator matrix to generate parity check bits;

storing the data words in a plurality of data buffer locations and the parity check bits in one or more parity check buffer locations of the buffer memory, the parity check buffer locations being different than the data buffer locations that contain the data;

reading the stored data and parity check bits;

regenerating the parity check bits; and

producing from the stored and regenerated parity check bits a result that is usable to directly identify

i. a data buffer location that contains a data word with ~~that contains~~ an erroneous bit, and

ii. the position of the erroneous bit in the data word.

25. (previously amended) A data storage system comprising:

a storage medium;

a controller coupled to the storage medium; and

a buffer memory coupled to the storage medium and the controller for storing data to be written to the storage medium and data read from the storage medium;

wherein the controller is operable to perform the following steps:

applying data to be stored in the buffer memory as a plurality of data words to a generator matrix to generate parity check bits, the generator matrix comprising a data portion and a parity check generation portion, and the parity check generation portion comprises rows of bits corresponding to binary representations of the respective data buffer locations to be used to store the data;

storing the data in the buffer memory at the data buffer locations; and

storing the parity check bits in the buffer memory in one or more parity check buffer locations that are different than the data buffer locations that contain the data.

26. (currently amended) An apparatus comprising:

a controller coupled to a storage medium; and

a buffer memory coupled to the controller for storing data to be written to the storage medium and data read from the storage medium;

wherein the controller is operable to perform the following steps:

applying data to be stored in a buffer memory as a plurality of data words to a generator matrix to generate parity check bits;

storing the data words in respective data buffer memory locations and the parity check bits in one or more parity check buffer locations that are different locations than the data buffer locations that contain the data;

reading the stored data and parity check bits;

regenerating the parity check bits; and

producing from the stored and regenerated parity check bits a result that is usable to directly identify

i. a data buffer location of a data word that contains an erroneous bit,

ii. and the position of the erroneous bit in the data word.

27. (currently amended) An apparatus comprising:

a controller coupled to a storage medium; and

a buffer memory coupled to the controller for storing data to be written to the storage medium and data read from the storage medium;

wherein the controller is operable to perform the following steps:

applying data to be stored in a plurality of addressable data buffer locations in the buffer memory to a generator matrix to generate parity check bits, the generator matrix comprising a data portion and a parity check generation portion, and the parity check generation portion comprises rows of bits corresponding to binary representations of the data buffer locations to be used to store the data;

storing the data in the buffer memory at the addressable data buffer locations; and

storing the parity check bits in the buffer memory in one or more addressable parity check buffer locations that have ~~are~~ different addresses~~locations~~ than the data buffer locations that contain the data.

28. (new) An apparatus comprising:

a controller coupled to a storage medium; and

a buffer memory coupled to the controller for storing data to be written to the storage medium and data read from the storage medium, the buffer memory including separately addressable multiple symbol storage locations;

wherein the controller is operable to perform the following steps:

applying data to a generator matrix to generate parity check bits, the generator matrix including a parity check generation portion that includes rows of bits that correspond to the addressable storage locations to be used to store the data;

storing the data in the corresponding addressable storage locations as a plurality of data words;

storing the parity check bits generated by the generator matrix in storage locations that are addressable separately from the storage locations in which the data words are stored;

retrieving the data words and the parity check bits from their respective storage locations and regenerating the parity check bits; and

producing from the stored and regenerated parity check bits a result that is usable to directly identify

i. the addressable storage location that contains a data word with an erroneous bit, and

ii. the position of the erroneous bit in the data word contained in the identified storage location.